LAMPIRAN A

Program Mikrokontroler

```
$regfile = "8535def.dat"
$crystal = 11059200
$baud = 9600

Dim Keypad As Byte
Dim Datakeypad As Byte
Dim Databit As String * 15
Dim Dataword As Word
Dim Jumdata As Byte

Config Lcdpin = Pin , Db4 = Portb.1 , Db5 = Portb.2 , Db6 = Portb.3 , Db7 = Portb.4 , E
= Portb.5 , Rs = Portb.6
Config Lcd = 16 * 2
Config Kbd = Portc , Debounce = 200

Cursor Off Noblink
Cls
Wait 1
Cls
Rem   1111111111111111
Lcd "   TUGAS AKHIR"
Wait 1
Lowerline
Lcd "    RESHANDARU"

Wait 2
Cls

Databit = ""
Jumdata = 0
Lcd "Data = "
Do
Keypad = Getkbd()
```

```
If Keypad < 16 Then
Datakeypad = Lookup(keypad , Tabel)

Cls
Jumdata = Jumdata + 1
Databit = Databit + Chr(datakeypad)
Lcd "Data = "
Lcd Databit
Waitms 500
If Jumdata = 3 Then
  Databit = Databit + chr(13)

  Print Databit
  Jumdata = 0
  Databit = ""
  Lowerline
  Lcd "  KIRIM DATA"
  Wait 2
  Cls
  Lcd "Data = "
End If

End If

Loop

End

Tabel:

Data &H31 , &H34 , &H37 , &H2A , &H32 , &H35 , &H38 , &H30 , &H33 , &H36 ,
&H39 , &H23 , &H41 , &H42 , &H43 , &H44
```

LAMPIRAN B

Program Delphi

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls, QCCom32, Buttons;

type
  TForm1 = class(TForm)
    Pdatamasuk: TPanel;
    Label1: TLabel;
    Label2: TLabel;
    Pdatamasukerror: TPanel;
    Timer1: TTimer;
    com: TQCCom32;
    Label3: TLabel;
    Pdataterkoreksi: TPanel;
    BitBtn1: TBitBtn;
    Label4: TLabel;
    GroupBox1: TGroupBox;
    RadioButton2: TRadioButton;
    RadioButton3: TRadioButton;
    Label5: TLabel;
    ps1: TPanel;
    ps3: TPanel;
    Label6: TLabel;
    Label7: TLabel;
    procedure Timer1Timer(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure RadioButton1Click(Sender: TObject);
    procedure RadioButton2Click(Sender: TObject);
    procedure RadioButton3Click(Sender: TObject);
```

```
    procedure modulo;
    procedure tertinggi;
    procedure kosongkanbuffer;
    procedure berierror;
    procedure tabelbinertoalfa;
    procedure tabelalfatobiner;
    procedure tabelbinernormaltoalfa;
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;
  posisi:byte;
  ketemu:boolean;
  data:array[0..14] of byte;
  dibagi:array[0..14] of byte;
  hasil:array[0..14] of byte;
  sisa:array[0..14] of byte;
  hasilXor:array[0..14] of byte;
  dataterkoreksi:array[0..14] of byte;
  x,xsisa,xhasil,xpembagi,xdibagi,posisitambah,posisiXor,posisiT:byte;
  M1,M3,dataSTR,S1STR,S3STR:string;

  datars232,datamasuk,datamasukerror,dataterkoreksi:string;

   s_in:byte;
   code,ts,clock:integer;
   m1_x,m3_x,s1_x,s3_x:integer;
   kpk:integer;
   jumerror,posrandom,poserror1,poserror2,jum:integer;
   total,binerdata,dataambil:string;
   Cx,Gx,Vx,tupple:string;
   A,A1,A2,A3,A23,BitPos1,BitPos2:byte;
   B1,B2,S3B0,S3B1,S3B2,S3B3,S3B4,bineralfa,S3alfa,S1alfa:byte;
```

```
implementation

{$R *.dfm}

procedure tform1.tabelbinernormaltoalfa;
begin
  if bineralfa=0 then A:=0;
  if bineralfa=1 then A:=1;
  if bineralfa=2 then A:=1;
  if bineralfa=3 then A:=4;
  if bineralfa=4 then A:=2;
  if bineralfa=5 then A:=8;
  if bineralfa=6 then A:=5;
  if bineralfa=7 then A:=10;
  if bineralfa=8 then A:=3;
  if bineralfa=9 then A:=14;
  if bineralfa=10 then A:=9;
  if bineralfa=11 then A:=7;
  if bineralfa=12 then A:=6;
  if bineralfa=13 then A:=13;
  if bineralfa=14 then A:=11;
  if bineralfa=15 then A:=12;
end;

procedure tform1.tabelbinertoalfa;
begin
   if tupple='0001' then A:=0;
   if tupple='0010' then A:=1;
   if tupple='0100' then A:=2;
   if tupple='1000' then A:=3;
   if tupple='0011' then A:=4;
   if tupple='0110' then A:=5;
   if tupple='1100' then A:=6;
   if tupple='1011' then A:=7;
   if tupple='0101' then A:=8;
   if tupple='1010' then A:=9;
   if tupple='0111' then A:=10;
```

```
  if tupple='1110' then A:=11;
  if tupple='1111' then A:=12;
  if tupple='1101' then A:=13;
  if tupple='1001' then A:=14;
end;

procedure tform1.tabelalfatobiner;
begin
  if A=0 then
     begin
      tupple:='0001';
      bineralfa:=0;
     end;
  if A=1 then
     begin
      tupple:='0010';
      bineralfa:=2;
     end;
  if A=2 then
     begin
      tupple:='0100';
      bineralfa:=4;
     end;
  if A=3 then
     begin
      tupple:='1000';
      bineralfa:=8;
     end;
  if A=4 then
     begin
      tupple:='0011';
      bineralfa:=3;
     end;
  if A=5 then
     begin
      tupple:='0110';
      bineralfa:=6;
     end;
```

```
if A=6 then
   begin
    tupple:='1100';
    bineralfa:=12;
   end;
if A=7 then
   begin
    tupple:='1011';
    bineralfa:=11;
   end;
if A=8 then
   begin
    tupple:='0101';
    bineralfa:=5;
   end;
if A=9 then
   begin
    tupple:='1010';
    bineralfa:=10;
   end;
if A=10 then
   begin
    tupple:='0111';
    bineralfa:=7;
   end;
if A=11 then
   begin
    tupple:='1110';
    bineralfa:=14;
   end;
if A=12 then
   begin
    tupple:='1111';
    bineralfa:=15;
   end;
```

```
  if A=13 then
     begin
      tupple:='1101';
      bineralfa:=13;
     end;
  if A=14 then
     begin
      tupple:='1001';
      bineralfa:=9;
     end;
end;


procedure tform1.berierror;
begin

    if jumerror=1 then
    begin
      repeat
        posrandom:=random(15);
        if posrandom=0 then posrandom:=random(15);
      until posrandom<>0;
      poserror1:=posrandom;
      datamasukerror:='';
      for jum:=1 to 15 do
      begin
        ketemu:=false;
        dataambil:=copy(datamasuk,jum,1);
        if jum<>poserror1 then
        begin
          datamasukerror:=datamasukerror+dataambil;
        end;
        if jum=poserror1 then
        begin
          if dataambil='1' then
             begin
               dataambil:='0';
               datamasukerror:=datamasukerror+dataambil;
```

```
          ketemu:=true;
        end;
      if (dataambil='0') and not ketemu then
        begin
          dataambil:='1';
          datamasukerror:=datamasukerror+dataambil;
        end;

    end;
    pdatamasukerror.caption:=datamasukerror;

  end;

end;
if jumerror=2 then
begin
  repeat
    posrandom:=random(15);
    if posrandom=0 then posrandom:=random(15);
  until posrandom<>0;
  poserror1:=posrandom;
  repeat
    posrandom:=random(15);
    if posrandom=0 then posrandom:=random(15);
  until (posrandom<>0) and (posrandom<>poserror1);
  poserror2:=posrandom;


  datamasukerror:='';
  for jum:=1 to 15 do
  begin
    ketemu:=false;
    dataambil:=copy(datamasuk,jum,1);
    if (jum<>poserror1) and (jum<>poserror2) then
    begin
      datamasukerror:=datamasukerror+dataambil;
    end;
    if (jum=poserror1) or (jum=poserror2) then
```

```
        begin
          if dataambil='1' then
            begin
              dataambil:='0';
              datamasukerror:=datamasukerror+dataambil;
              ketemu:=true;
            end;
          if (dataambil='0') and not ketemu then
            begin
              dataambil:='1';
              datamasukerror:=datamasukerror+dataambil;
            end;
        end;
      end;
      pdatamasukerror.caption:=datamasukerror;

    end;

end;

procedure tform1.kosongkanbuffer;
begin
 for posisi:=0 to 14 do
 begin
   data[posisi]:=0;
   dibagi[posisi]:=0;
   hasil[posisi]:=0;
   sisa[posisi]:=0;
   hasilXor[posisi]:=0;
 end;

end;


procedure tform1.tertinggi;
begin
{pencarian dibagi tertinggi}
ketemu:=false;
```

```
for posisi:=14 downto 0 do
begin
  if (dibagi[posisi]=1) and not ketemu then
  begin
     Xdibagi:=posisi;
     ketemu:=true;
  end;
end;

end;

procedure tform1.modulo;
begin
For posisi:=14 downto 0 do
begin
    if data[posisi]=1 then
    begin

       Xpembagi:=posisi;
       if Xpembagi>=Xdibagi then
       begin
         Xhasil:=Xpembagi-Xdibagi;

          Hasil[Xhasil]:=1;

         for x:=0 to 14 do sisa[x]:=0;
         for posisitambah:=14 downto 0 do
         begin
           if dibagi[posisitambah]=1 then
           begin
             Xsisa:=Xhasil+posisitambah;
             Sisa[xsisa]:=1;
           end;
         end;

         for posisiXor:=14 downto 0 do
         begin
           hasilXor[posisiXor]:=data[posisiXor] xor sisa[posisiXor];
```

```
          data[posisiXor]:=hasilXor[posisiXor];
        end;

      end;
    end; {data[posisi]=1 }
end;
end;

procedure TForm1.Timer1Timer(Sender: TObject);
begin

{--------------------------------Baca Data dari COM komputer---------------------------------}
datars232:=com.read;

if length(datars232)>1 then
begin
kosongkanbuffer;
  datars232:=copy(datars232,1,3);
  {caption:=datars232;}
  Total:='';

  val(datars232,X,code);
  repeat
   Total:=inttostr(X mod 2)+Total;
   X:=X div 2;
  until X=0;

  if length(total)=7 then datamasuk:=total;
  if length(total)=6 then datamasuk:='0'+total;
  if length(total)=5 then datamasuk:='00'+total;
  if length(total)=4 then datamasuk:='000'+total;
  if length(total)=3 then datamasuk:='0000'+total;
  if length(total)=2 then datamasuk:='00000'+total;
  if length(total)=1 then datamasuk:='000000'+total;

  Cx:=datamasuk+'00000000';
  Gx:='000000111010001';
```

```
  For x:=14 downto 0 do
  begin
    if copy(Cx,15-x,1)='1' then  Data[x]:=1 else Data[x]:=0;
    if copy(Gx,15-x,1)='1' then  Dibagi[x]:=1 else Dibagi[x]:=0;
  end;
  Tertinggi;
  Modulo;
  Modulo;
  dataSTR:='';
  for x:=14 downto 0 do
  begin
   dataSTR:=dataSTR+inttostr(data[x]);
  end;

  Vx:=datamasuk+copy(dataSTR,8,8);

  datamasuk:=Vx;
  pdatamasuk.caption:=Vx;
  Berierror;

{--------- Pencarian Sindrom S1 ---------}
  pdatamasukerror.Caption:=datamasukerror;
  Kosongkanbuffer;
  Vx:=pdatamasukerror.caption;
  M1:='000000000010011';
  M3:='000000000011111';

  For x:=14 downto 0 do
  begin
    if copy(Vx,15-x,1)='1' then  Data[x]:=1 else Data[x]:=0;
    if copy(M1,15-x,1)='1' then  Dibagi[x]:=1 else Dibagi[x]:=0;
  end;
  Tertinggi;
  Modulo;
  dataSTR:='';

  for x:=14 downto 0 do
```

```
  begin
    dataSTR:=dataSTR+inttostr(data[x]);

  end;
  S1STR:=copy(dataSTR,12,4);
  Ps1.caption:=S1STR;

{--------- Pencarian Sindrom S3 ---------}
  Kosongkanbuffer;
  Vx:=pdatamasukerror.caption;
  M1:='000000000010011';
  M3:='000000000011111';

  For x:=14 downto 0 do
  begin
     if copy(Vx,15-x,1)='1' then  Data[x]:=1 else Data[x]:=0;
     if copy(M3,15-x,1)='1' then  Dibagi[x]:=1 else Dibagi[x]:=0;
  end;

  Tertinggi;
  Modulo;
  dataSTR:='';
  for x:=14 downto 0 do
  begin
    dataSTR:=dataSTR+inttostr(data[x]);
  end;
  S3STR:=copy(dataSTR,12,4);
  Ps3.caption:=S3STR;

  if jumerror=1 then
  begin
    tupple:=S1STR;
    tabelbinertoalfa;
    B1:=A;
  end;
```

```
  S3B3:=0;
  S3B2:=0;
  S3B1:=0;
  S3B0:=0;
  if jumerror=2 then
  begin

{=---------------}

  tupple:=S1STR;
  tabelbinertoalfa;
  S1alfa:=A;

   S3B3:=0;
   S3B2:=0;
   S3B1:=0;
   S3B0:=0;

   if copy(S3STR,1,1)='1' then S3B3:=9;
   if copy(S3STR,2,1)='1' then S3B2:=6;
   if copy(S3STR,3,1)='1' then S3B1:=3;
   if copy(S3STR,4,1)='1' then S3B0:=1;

   A:=S3B3;
   tabelalfatobiner;
   S3B3:=bineralfa;

   A:=S3B2;
   tabelalfatobiner;
   S3B2:=bineralfa;

   A:=S3B1;
   tabelalfatobiner;
   S3B1:=bineralfa;

   S3alfa:=S3B3 + S3B2;
   S3alfa:=S3alfa + S3B1;
   bineralfa:=S3alfa and $0F;
```

```
    tabelbinernormaltoalfa;
    S3alfa:=A;
{--------error locator---------}
       A1:=S1alfa;
       A2:=S1alfa*2;
       A3:=S3alfa-A1;

       A:=A2;
       tabelalfatobiner;
       A2:=bineralfa;

       A:=A3;
       tabelalfatobiner;
       A23:=bineralfa;

       A23:=A23 xor A2;
       bineralfa:=A23 and $0F;
       A:=bineralfa;
       tabelalfatobiner;

{------ Koreksi data ------}
      for x:=14 downto 0 do
      begin
        if poserror1=x then
           datateRkoreksi[x]:=datamasukrror XOR B1;
        if poserror2=x then
           datateRkoreksi[x]:=datamasukrror XOR B2;
      pdatateRkoreksi.caption:=datateRkoreksi;

  end;
end;
end;

procedure TForm1.FormCreate(Sender: TObject);

begin
com.pick;
datamasuk:= '';
```

```
datamasukerror:=";

pdatamasuk.caption:=     ";
pdatamasukerror.caption:=";
ps1.caption:=            ";
ps3.caption:=            ";

jumerror:=0;

end;

procedure TForm1.RadioButton1Click(Sender: TObject);
begin
jumerror:=0;

end;

procedure TForm1.RadioButton2Click(Sender: TObject);
begin
jumerror:=1;

end;

procedure TForm1.RadioButton3Click(Sender: TObject);
begin
jumerror:=2;

end;

end.
```